

DUAL CONGRUENT INTEGER CRYPTOGRAPHIC ALGORITHM TO SECURE SENSITIVE INFORMATION TRANSIENT

YAHIA S. AL-HALABI

Professor, Princess Sumaya University for Technology, Amman, Jordan,

ABSTRACT

Cryptography is one of the most important tools that provide hiding data and information security usually done through mathematical manipulation of the data with an incomprehensible format for unauthorized users. It has long been used by government, public and private companies and militaries too to facilitate secret communication. Mainly, it is commonly used within many kinds of civilian systems in protecting information. Building and creating a key that would prevent an unauthorized decryption of message is the challenging part of cryptography. This paper aimed at providing algorithms to secure sensitive information transit via unsecure channels from the hands of the Internet criminals by scrambling the sensitive information using consecutive mathematical-based algorithm which performs dual ciphered and re-ciphered operation (Dual encryption), then dual decryption and re-decryption, by principles of congruent, for both consecutive encryption and decryption schemes (Dual decryption). Additionally, the use of basic mathematical characteristics of integer numbers generators applied in both steps in dual encryption and dual decryption which are the novel of our project.

The suggested algorithm was tested with samples of real data and the result practically demonstrated how dual schemes are indeed of relevance in providing strong and complex decryption keys to protect sensitive information. The dual consecutive schemes proved that more illegal trials attempted, the more difficult it is for the criminals or hackers to unveil the sensitive information. The results of this algorithm shows that discourage the criminals will be based on the complexity of the dual encryption and decryption key. This paper is part of a commercial big project considered as a first step to be extended to use finally randomized integers during encryption process which will never be predicted and will be applied as a first step by using different distinct or random process during encryption and decryption processes.

KEYWORDS: Cryptographic Algorithms, Cryptology, Cryptanalyst, Decryption, Encryption, Cipher, Protocol, Data Security.

INTRODUCTION

Security of data and information has become a very critical aspect of modern computing systems. Cryptography which is the science of secrecy is the practice and study of hiding information. Modern Cryptography intersects the disciplines of mathematics, Computer Science and Engineering with different applications including computer passwords, electronic commerce, ATM cards and others, and its goal is the protection of data and information confidentiality with cryptosystems support [1, 3, 5, 10]. It is composed by three elements: a cryptographic algorithm, a keys generation system and a protocol for keys distribution. There have been several high-profile incidents of loss of individual privacy and confidentiality of data and information in unsecure communication channels. If no security measures are taken, then no

doubt that such data and other sensitive information will continuously be faked and formatted by the hackers [7, 16]. Cryptographic algorithms which help prevent interception and enhance message security are now of primary importance. The challenging part of cryptography today is how to create a key that would prevent an unauthorized decryption of message and key is simply a parameter to the algorithm that allows the encryption and the decryption processes to occur.

In cryptography, encryption is the process of transforming information using an algorithm (called a cipher) to make it unreadable to anyone except those possessing special knowledge, usually referred to as a key. The result of the process is encrypted information (in cryptography, referred to as cipher text). The reverse process, i.e., to make the encrypted information readable again, is referred to as decryption (i.e., to make it unencrypted). Encryption software executes an algorithm that is designed to encrypt computer data in such a way that it cannot be recovered without access to the key. Software encryption is a fundamental part of all aspects of modern computer communication and files protection to prevent third parties from recovering the original information. This is particularly important for sensitive data. Encryption, by itself, can protect the confidentiality of messages, but other techniques are still needed to protect the integrity and authenticity of a message.

Different techniques developed by different researcher to protect network traffic, personal data, and corporative IT infrastructure, [1, 2, 3]. Improved Cryptanalysis of researchers [8, 12, 16] is also developed to enhance the well known technique [1, 5]. U.S.A Government in 2003, announced that AES (Advanced Encryption Standard) may be used to protect classified information: the cipher strength of all key lengths of AES are sufficient to protect classified information up to the SECRET level, however, TOP SECRET information requires use of either 192 or 256 bit keys [5, 7, 15]. However, recent paper [6, 9, 12] claims that the 10-round AES is theoretically possible to crack by cryptanalysis [7, 14]. Crypto algorithms also discussed the primary challenge in providing security in mobile devices is minimizing energy consumption and maximizing security [3]. Scalable features such as scalable key establishment protocols and scalable authentication schemes, in which different security, performance and energy trade-offs are enabled for different application scenarios are especially desirable [4, 6, 12]. Enhancement of Security through a Cryptographic Algorithm Based on Mathematical representation is also discussed [5, 11, 6] and other work by Microsoft® provides a .NET framework technology that has a crypto service provider for information encryption/ decryption on a handheld PC with DES, 3DES, AES, RC2 algorithms [8, 9, 13]. Researchers usually, study the algorithms based on such cryptography based on international standard such as DES, RSA and others in order to develop new improved Cryptography algorithm or enhanced one [2, 8, 11].

From basics of information security, the well known cipher DES (the Data Encryption Standard) is a symmetric block cipher developed also by IBM. The algorithm uses a 56-bit key to encipher/decipher a 64-bit block of data. The key is always presented as a 64-bit block, every 8th bit of which is ignored. The algorithm is best suited to implementation in hardware, probably to discourage implementations in software, which tend to be slow by comparison. However, modern computers are so fast that satisfactory software implementations are readily available. RSA is a public key algorithm. The key used for encryption is different from (but related to) the key used for decryption.

The algorithm is based on modular exponentiation. Numbers (e), (d) and (N) are chosen with the property that if (A) is a number less than (N), then:

$$(A * e \bmod N) d \bmod N = A \quad (1)$$

In other words, one can encrypt (A) with (e) and decrypt using (d). Conversely you can encrypt using (d) and decrypt using (e). Accordingly, one can say:

- The pair of numbers (e, N) is known as the public key and can be published.
- The pair of numbers (d, N) is known as the private key and must be kept secret.

The number e is known as the public exponent, the number (d) is known as the private exponent, and (N) is known as the modulus. When talking of key lengths in connection with RSA, what is meant is the modulus length. An algorithm that uses different keys for encryption and decryption is said to be asymmetric.

Any knowledge about public key can be used to create encrypted messages, but only the owner of the secret key can decrypt them. Conversely the owner of the secret key can encrypt messages that can be decrypted by anybody with the public key. Anybody successfully decrypting such messages can be sure that only the owner of the secret key could have encrypted them. This fact is the basis of the digital signature technique. Without going into detail about how (e), (d) and (N) are related, (d) can be deduced from (e) and (N) if the factors of (N) can be determined. Therefore the security of RSA depends on the difficulty of factorizing (N). Because factorization is believed to be a hard problem, the longer (N) is the more secure the cryptosystem. Given the power of modern computers, a length of about 1024 bits is recommended and it is considered reasonably safe, especially for serious commercial use. Depending on the previous known techniques RSA and DES, it is well known that the problem with choosing long keys is that RSA is very slow compared with a symmetric block cipher such as DES, and the longer the key the slower it is. Accordingly, the best solution is to use RSA for digital signatures and for protecting DES keys. Bulk data encryption should be done using DES.

This paper provides cryptographic algorithms using mathematical formulations that enciphered message into a form of dual consecutive encryptions and dual consecutive decryption (which will be used in the future for our big project by applying random generator that are deciphered through random generator set operation followed by dual consecutive methods to decrypt the message).

BASIC INFORMATION

There are several factors that affect the choice of an encryption algorithm including speed and security. The simplest method for cipher would be an **XOR** operation defined as : with a constant value **K**, of each byte of plain text **D**, to produce a cipher value **S**,

$$D \text{ XOR } K = S \quad (2)$$

Ciphers can be categorized into two general types: public key ciphers and symmetric key ciphers. Public key systems are based upon algorithms that are at least strongly believed to be "one-way" operations. That is, encryption with one member of a key pair is only easily reversed (decrypted) using the other member of the pair. Further, one member of the pair (the public key) cannot be easily used to determine the other (the private key). Provided the problems posed by the system to cryptanalyst are effectively unsolvable and the system is effectively secure. Such systems are used for key exchanges (for subsequent use of symmetric key ciphers), digital signatures and others.

Key generation has two phases. The first phase is a choice of **algorithm parameters** which may be shared between different users of the system, while the second phase computes public and private keys for a single user. In summary, encryption and decryption process is shown in Figure 1.

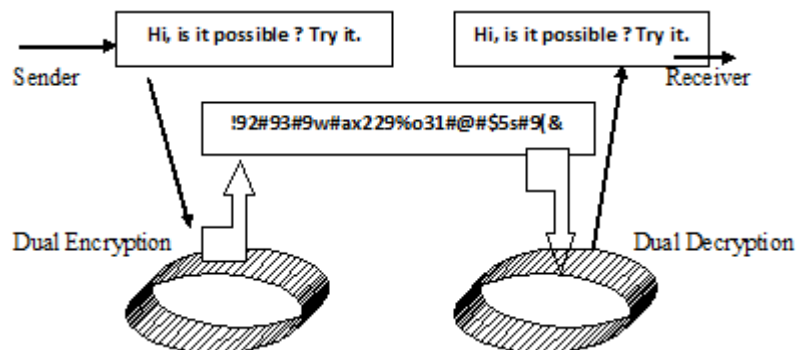


Figure 1: Structure of Dual Encryption and Dual Decryption Process

There are many types of data encryption algorithm but not all are reliable. Advanced Encryption Algorithm (AES) is the new standard used by many data encryption software worldwide and is considered as the most secure and reliable algorithm. AES allows maximum of 256-bits. Encryption algorithm is considered to be practically secure and reliable technique to protect your data. Although there are theoretical attacks and some theoretical weaknesses in the cipher, but they are unfeasible to mount in practice. In the recent years with the advancement of technology, old ciphers have also been replaced by the Advanced Encryption Standard (AES). Furthermore, Data Encryption Standard (DES) has also been backed away as a standard by the National Institute of Standards and Technology (National Bureau of Standards). There are some cases where a key is required to be developed specifically for the use of some special persons, while on the other hand some encryption protocols are standardized so that people can easily communicate with each other without requiring any special key.

THE PROPOSED ALGORITHM

Data encryption considered the best approach to protect your confidential information because it transform information into an unreadable format that cannot be read without knowing the special key or a password.

Our algorithm is defined by two (dual or consecutive) steps for data encryption in order to satisfy the above requirement, and also other dual steps for decryption. Data encryption in our algorithm is a process used in cryptography to define the term of transforming data or information into an unreadable and inaccessible to anyone format by using a cipher built by different operations. Normally, the Cipher text is obtained using the following steps:

Construct random distinct key for information, to make it inaccessible to anyone who does not know your correct form. [By availability of high power of computer hardware and fast computer software, one can predict the possible solution of the key easily]. This step of the algorithm is trivial step applied in all available Cipher techniques. Construct the first step of the dual algorithm based on characteristics of integer and natural numbers, which generate a new Cipher for the above step to generate new keys which we claimed, not possibly predicted. The dual operation applied to strength encrypts the data using the same algorithm, to grantee that reconstruction another encryption yields to complicated form. [But as we said in the above step (by availability of high power of computer hardware and fast computer software and its base on

mathematical formula)], then cryptographs are might possibly able to solve the solution of the key too, using dual encryption but still not easy to discover for its complexity as values.

The key obtained in the last step, is ciphered again, using the characteristics of integer numbers. This dual step of encryption, use any distinct normal numbers or distinct random numbers to be added to keys obtained in previous step to generate new keys for encryption which is: key-trivial, mathematical formulation, distinct integers or distinct random integer number.

Both steps, generate hardly or even absolutely impossible to generate information about original key, since using integers or distinct integers in the first and second dual strength step of the algorithm cannot be identified easily by hackers and as what we said before:

“by availability of high power of computer hardware and fast computer software”, one cannot be able to predict the possible solution of the key, easily. This result will be used later in next proposed project to generate random integers in both steps of encryption and decryption processes which will be impossible to predict.

MATHEMATICAL BACKGROUND AND ALGORITHM

In this section, we will present basic mathematical equations and method of solution which is shown to be simple to construct, easy to solve and trivial to be constructed.

Initially, consecutive numbers (or more properly, consecutive integers) are integers n_1 and n_2 such that: $n_2 - n_1 = 1$, i.e., n_2 follows immediately after n_1 .

Given two consecutive numbers, one must be even and one must be odd by definition, since the difference between any two consecutives is one. Since the product of an even number and an odd number is always even, the product of two consecutive numbers (and, in fact, of any number of consecutive numbers) is always even.

To investigate several sets of three consecutive numbers n_1, n_2, n_3 , one can notice that the square of n_2 which is the middle, is always one more than the product of the other two numbers n_1 and n_3 . This is one of the special characteristics for natural numbers like 6 as an example, and other similar consecutive numbers. We are concerned only with this characteristic and its relations with cubic equations, and otherwise different relation can be obtained too for different generating new algorithms similar to the one we propose. Also, if we start with an even number and each number in the sequence is 2 more than the previous number then we will get consecutive even integers, as an example 16,18,40 and if we start with an odd number and each number in the sequence is 2 more than previous number then we will get consecutive odd integers, (as an example 33, 35, 37,.....), and such cases are important for different schemes of the governing equations under study. Also, any positive integer n (as an example the number 6), is called perfect number such that:

$$s(n)=n \quad (3)$$

where $s(n)$ is restricted divisor function (i.e., the sum of proper divisor of n), or equivalently:

$$\sigma(n) = 2n \quad (4)$$

where $\sigma(n)$ is the divisor function (i.e., the sum of divisors of n including n itself). For example, the first few perfect numbers are 6, 28, 496, 8128 ... The first perfect integer number 6 has a special characteristic for natural integers such that: $6 = 1 + 2 + 3$ and $6 = 1 * 2 * 3$ as consecutive integers applied for addition or multiplication. Similar can be applied for other consecutive integers, in addition to even and odd consecutives with different properties. Still, many open problems related to such proposed cases. For the purpose of our research, cubic equation, with terms $(x-1)$, (x) , $(x+1)$ which are consecutive, (or of the form (x) , $(x+1)$, $(x+2)$), can be written in the following form:

$$[(x-1) * (x) * (x+1)] / 6 = k \quad (5)$$

for all $x > 1$, and k is an integer. This is valid for any three consecutive numbers of the form $(x-1)$, (x) and $(x+1)$ or of a form (x) , $(x+1)$, $(x+2)$ for x greater than zero, or for any consecutive integers of the form $(x-i)$, (x) , $(x+i)$ for any integer $i < x$.

ALGORITHM OF ENCRYPTION AND DECRYPTION

Step-1

Sender will develop an arbitrary key for all characters in the message, called x taking in consideration that they are integers and distinct (not necessarily prime). This step can be generalized automatically. It is called code assignment.

Step-2

Apply equation (5), to end up with an arbitrary integer key called (K) as initial encryption of a certain character in the message, by evaluating the value of K as:

$$K = [(x-1) * (x) * (x+1)] / [6] \quad (6)$$

Step-3

Construct a new integer Key called (M) , based on (K) , different than the original (x) for a certain character. Such repetition of dual encryption gives the value of M where M is equal to:

$$M = [(K-1) * (K) * (K+1)] / [6] \quad (7)$$

which generates the dual encryption step? This dual step of encryption generates different distinct normal numbers to be generated by keys obtained in previous step to form new keys for such strong dual encryption as if it is a new key-trivial, using mathematical formulation based on idea of distinct integers. This procedure is similar to the well known mathematical notation of applying composite functions:

$$M = g(f(x)) \quad (8)$$

This procedure is similar to the well known mathematical notation of applying composite functions:

$$K = f(x), \text{ and } M = g(K), \text{ and } f(x) = [(x-1) * (x) * (x+1)] / [6] \quad (9)$$

and x is integer distinct key for information, claimed to be inaccessible to anyone who does not know this key in order to use it to create encrypted messages, but only the owner of the secret key can decrypt them. By the end of this step, encryption is completed.

Step-4

Decryption is a reverse operation for encryption process. Root of the cubic equation, with solution M_x is calculated as follows:

$$M_x = [[(M-1)*(M)*(M+1)]^{(1.0/3.0)} + 1 \quad (10)$$

Step-5:

Dual decryption is obtained using the same operation, by evaluating the solution K_x as follows:

$$K_x = [[(M_x-1)*(M_x)*(M_x+1)]^{(1.0/3.0)} + 1 \quad (11)$$

The code K_x in this step is a result of the dual decryption for code M_x , which is also the result of first encrypting step. The value of K_x will be equivalent to the cipher key (initial code x).

APPLICATION OF THE ALGORITHM

Let $\{ (x(i), c(i)) \}, i = 0, \dots, j$ be the set of pair of original cipher keys which are arbitrary, distinct used as first trivial encryption generation process. This step can be constructed as a first level of automatic encryption. (Code assignment for specific character). Assume character "d" is ciphered to a value 4 then the value of the expression $(x-1)*x*(x+1)/6$ gives an encryption for "d" to be: $(64-4)/6=10$ as a first step. Repeat the same and perform dual step of encryption. Accordingly, the new secured encrypted integer key will be: $[(10)^3-10]/6$ which is equal to 165.

For decryption step, one has to find the actual key by factorization of the sent code 165 as a product of $(x-1)*(x)*(x+1)$ or as $(x)*(x+1)*(x+2)$ or other similar sequence of terms satisfy the integer characteristics. This gives the value of original key. The following are partial main C++ similar codes for both dual encryption and decryption steps:

```
/* Define the array of cipher c[i], x[i], i=0.. n [given number of characters required*/
for (i=32;i<n;i++) /* Given n : number of different codes, cipher codes x[i] */
{
    d1[i]=(x[i]-1) * x[i] * (x[i]+1)/6;

    /* Example d1[i] = 10 for character "d" for initial cipher code x[i] 4 for "d" */

    /* Second dual encryption of dd[i] */
    dd1[i]=(d1[i]-1)*d1[i]*(d1[i]+1)/6;

    /*dd1[i] = will be sent for decryption */
}

/* character "d" with cipher 4 encrypted initially to 10 then secondly as 165 */
/* encryption step-- dd1[i] will be sent for first step of decryption. */
/* Now, if the final encryption code equal 165, then: */
/*       $M_x = (\text{Smallest Integer } [ [ (M-1)*(M)*(M+1)]^{(1.0/3.0)} ] + 1$  */
```

```

/* is obtained using the same reverse operation. */
/* Secondly, evaluate the solution of Kx as: */
/*      Kx = (Smallest Integer [ (Mx-1)*(Mx)*(Mx+1)]**(1.0/3.0)) +1 */
/* User should be careful in assigning type of parameters for Smallest integers. */

for (i=32;i<n;i++)
{
    xd2[i]=(unsigned long long) pow((double long) (dd1[i]*6.0),(1.0/3.0))+1;

    /* The same for decryption of xd2[i], save it in xd1[i] */
    xd1[i]=(unsigned long long)pow((double long)(xd2[i]*6),(1.0/3.0))+1;

    /* xd1[i]= 3 + 1  which is equal  4 for the example above for character "d" . */
    /* -----Second step of decryption is Finished----- */

```

RESULTS AND CONCLUSIONS

The proposed algorithm can be used to generate a plenty of other algorithms based on the theory described above. Researchers can easily derive new form or equations allow them to construct other formulas. The challenging part of such proposed cryptography algorithm or other derived ones, concentrates on how to create keys that would prevent an unauthorized decryption of message and keys are simply a parameter to the algorithm that allows the encryption and the decryption processes to occur. The idea of this algorithm shows that resources required for revealing a secret message should be strong and complex enough through a hiding keys. That was obtained by the application of dual steps of both encryption and decryption steps. It is clearly observed from the result of the proposed algorithm that any method of attack to find the decryption key by the unauthorized user required of not only the original cipher step but also the solution of complex formula based on composite functions of solving the set of decryption of complex big and large integer cubic unpredictable equations, twice, for both dual encryption and decryption. Representation of all ASCII codes of all character set in the different numbering systems is selected. The first process of the algorithm is to build a cipher codes and construct the set of pairs $(x[i], c1[i])$ as a hidden keys for cryptography. Simply, we assigned $x[i], c1[i]$ as the values associated with the variables index in the APPENDIX-1, formulating the equation and understanding the position of the alphanumeric characters in the attached APPENDIX-1. Hence, by applying the algorithm proposed, one can notice that the proposed method provides strong protection of data from unauthorized access in storage or transmission. This is in agreement with an earlier study by other researchers who revealed that resources required for revealing a secret message should be strong and complex enough through a hiding key.

The attached APPENDIX-1, shows code representation of all character set. Assuming that cipher selected as follows: $x_i = x[i] = c1[i]$, for all $i=32, \dots, 127$. For the purpose of simulation, assume that the actual message to be encrypted is as follow:

God Bless You--- ---2012 - 2014-END

According, the initial cipher $x_i=x[i]$, $i=1 \dots m=37$ [the length of the strength], will be generated as shown in Table 1.

Table 1: Cipher Codes, The Correct Locations for the Sample Message

$x_0=71$	$x_1=111$	$x_2=100$	$x_3=32$	$x_4=66$	$x_5=108$	$x_6=101$	$x_7=115$
$x_8=115$	$x_9=32$	$x_{10}=89$	$x_{11}=111$	$x_{12}=117$	$x_{13}=45$	$x_{14}=45$	$x_{15}=45$
$x_{16}=32$	$x_{17}=32$	$x_{18}=32$	$x_{19}=45$	$x_{20}=45$	$x_{21}=45$	$x_{22}=50$	$x_{23}=48$
$x_{24}=49$	$x_{25}=50$	$x_{26}=32$	$x_{27}=45$	$x_{28}=32$	$x_{29}=50$	$x_{30}=48$	$x_{31}=49$
$x_{32}=52$	$x_{33}=45$	$x_{34}=69$	$x_{35}=78$	$x_{36}=68$			

The initial process of dual encryption of any message is applied accordingly, and results are shown in details in Table 2.

Table 2: General Code of Every Character Set (Dually Encrypted, Generated Simultaneously) as Pairs of ($d1[i]$, $dd1[i]$, $i=32 \dots 95$), Sequentially

i	($d1[0+i]$, $dd1[0+i]$)	($d1[1+i]$, $dd1[1+i]$)	$d1[2+i]$, $dd1[2+i]$)	($d1[3+i]$, $dd1[3+i]$)
32 - 35	5456 27068975560	5984 35712766320	6545 46728053680	7140 60665722810
36 - 39	7770 78182904205	8436 100059529570	9139 127216890580	9880 160738377020
40 - 43	10660 201892580890	11480 252158963420	12341 313256292580	13244 387174069590
44-47	14190 476207174135	15180 582993969470	16215 710558120360	17296 862354388840
48-51	18424 1042318685100	19600 1254922663400	20825 1505233165800	22100 1798976829650
52-55	23426 2142610188225	24804 2543395607610	26235 3009483416940	27720 3550000603380
56-59	29260 4175146457790	30856 4896295571860	32509 5726108602620	34220 6678651235630
60-63	35990 7769521793835	37820 9015987955030	39711 10437133057120	41664 12054012486880
64-67	43680 13889820664720	45760 15970069155040	47905 18322776449120	50116 20978669985130
68-71	52394 23971400987765	54740 27337772728210	57155 31117982823620	59640 35355880214060
72-75	62196 40099237473890	64824 45400039133900	67525 51314786710100	70300 57904821154950
76-79	73150 65236663466975	76076 73382374215150	79079 82419932755160	82160 92433636935640
80-83	85320 103514524113780	88560 115760814321240	91881 129278376443160	95284 144181218295170
84-87	98770 160592001505705	102340 178642582133610	105995 198474577973980	109736 220239963528420
88-91	113564 244101693639430	117480 270234356812420	121485 298824859272940	125580 330073140831070
92-95	129766 364192923649555	134044 401412495037190	138415 441975525414160	142880 486141922621520

Table 3 shows the dual decryption process for the sample message under investigation, [by receiving the encrypted codes dd1[i] (\rightarrow Sent, encrypted) of the message], and generating an ordered pairs (xd2[i]-first decryption received, xd1[i]-second decryption), as a final stage which yields to an accurate decryption codes after final description [Generated decrypted code].

Table 3: General Codes Dually Decryption Process Generated Simultaneously as Pairs (Xd2 [i], Xd1 [i], I=0....36)

i	Actual Message	dd1[i] \rightarrow Sent, Encrypted	xd2[i] First	xd1[i] Second	Generated Decrypted Code	Actual Generated Message, Space=' '
0	G	35355880214060	59640	71	G	G
1	o	1973313369476680	227920	111	o	o
2	d	771373479909725	166650	100	d	d
3		27068975560	5456	32	space	space
4	B	18322776449120	47905	66	B	B
5	l	1542045157297095	209934	108	l	l
6	e	843644802138050	171700	101	e	e
7	s	2713795020247090	253460	115	s	s
8	s	2713795020247090	253460	115	s	s
9		27068975560	5456	32	space	space
10	Y	270234356812420	117480	89	Y	Y
11	o	1973313369476680	227920	111	o	o
12	u	3421680775889740	273819	118	u	u
13	-	582993969470	15180	45	-	-
14	-	582993969470	15180	45	-	-
15	-	582993969470	15180	45	-	-
16		27068975560	5456	32	space	space
17		27068975560	5456	32	space	space
18		27068975560	5456	32	space	space
19	-	582993969470	15180	45	-	-
20	-	582993969470	15180	45	-	-
21	-	582993969470	15180	45	-	-
22	2	1505233165800	20825	50	2	2
23	0	1042318685100	18424	48	0	0
24	1	1254922663400	19600	49	1	1
25	2	1505233165800	20825	50	2	2
26		27068975560	5456	32	space	space
27	-	582993969470	15180	45	-	-
28		27068975560	5456	32	space	space
29	2	1505233165800	20825	50	2	2
30	0	1042318685100	18424	48	0	0
31	1	1254922663400	19600	49	1	1
32	4	2142610188225	23426	52	4	4
33	-	582993969470	15180	45	-	-
34	E	27337772728210	54740	69	E	E
35	N	82419932755160	79079	78	N	N
36	D	23971400987765	52394	68	D	D

From such compound processes of initial cipher, the dual encrypted and dual decrypted messages, will not visible to an unauthorized person without the decryption keys. The algorithm of this model and its simple implementation and its complicating process of trying to visible any message by any unauthorized person is also makes this algorithm a novel one. The result in Table 2 and 3 have proved the validity, reliability and practicality of the proposed application (encryption and

decryption algorithms) in handling real life situations in an unsecured environment. It can be used to stimulate numerous applications in the area of Computer Science, Banking Security processes, Data Transfer Media, Store and Encrypt Data Storage, different topics in Science and Engineering too. Relationship between execution times and text lengths are showed in Figure 2. It is clear that the execution time is increased as encryption and decryption processes of the length of the message are increased too. This algorithm can be applied easily on all type of text files of any sizes, and can be expanded by user using other formulas or composite formulas for both encryption and decryption, to prevent any attack by the unauthorized users twice, for both dual encryption and decryption. Whatever the input data is paired with any cipher code associated with any distinct integer value, then the proposed algorithm is able to encrypt and decrypt easily.

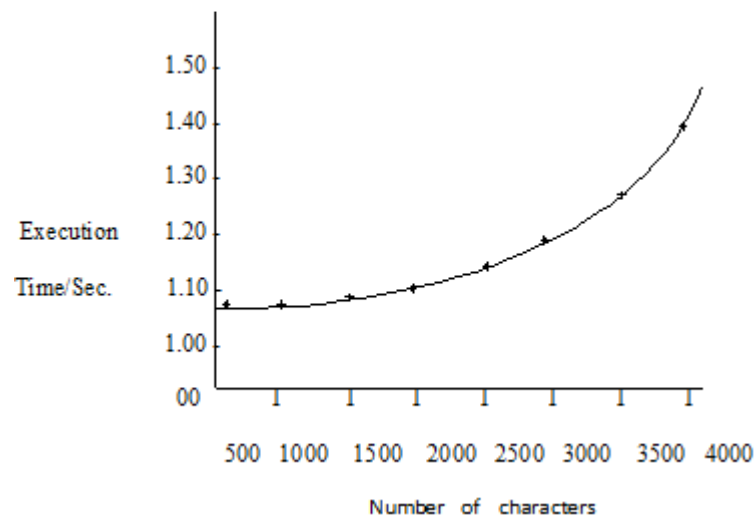


Figure 2: Relationship between Execution Times and Message Lengths

The proposed algorithm ensured that unauthorized users do not have access to sensitive documents stored or transmitted to other sides without knowing the actual secret key in the cipher process, and by using exactly such fixed mathematical formulas which yields to generate complicated encryption integers and reverse process for decryption. Users can think of new similar formulas which guarantee sending deferent codes encrypted for different receivers but the final result of decryption process will be the same for original message which is usually used in military communications.

ACKNOWLEDGEMENTS

This work was supported by the Princess Sumaya University for Technology, during my sabbatical leave from King Hussain School for Computing-Computer Science Department during the academic year 2013-2014.

REFERENCES

1. A. Baums, Energy consumption optimization in hard real- time system CMOS processors // *Electronics and Electrical Engineering*. – Kaunas: *Technologija*, 2006. – No. 4(68). – P.19–22
2. A. Biryukov, N. Keller, D. Khovratovich, A. Shamir A. Key recovery attacks of practical complexity on AES-256 variants with up to 10 rounds // *Advances in Cryptology* – 2008, edition, 2001.
3. A. Raluca, H. Frank, Z. Nickolai. An Ideal-Security Protocol for Order-Preserving Encoding. *IEEE S&P/Oakland 2013 (IEEE Symposium on Security and Privacy)*. Extended paper in Cryptology e Print Archive, 2013/129.

4. D. Boneh, *On the importance of Checking Cryptographic Protocols for Faults*, *Journal of Cryptology*, Springer-Verlag, Vol. 14, No. 2, pp. 101-119, 2001.
5. E. Simion, V. Preda, A. Popescu, *Cryptanalysis. Mathematical Techniques and Methods*. University of Bucharest Publishing House, 2004.
6. Eurocrypt *Int. Conf. on the Theory and Applications of Cryptographic Techniques*, 2010, 29th Lecture Notes in Computer Science, 2010. – Vol. 6110. – P. 299–319.7.
7. G. Crina, & G. Ajith A new approach for solving non-linear equations system. *IEEE Transaction on Systems, Man, and Cybernetic*, (2008).38(3).
8. G. Kessler, *Handbook on local area networks: An overview of cryptography*. United Kingdom: Auerbach. (2010). Retrieved from <http://www.garykessler.net/library>.
9. H. Tilborg, *Fundamentals of Cryptology*, *Kluwer Academic Publisher*, second
10. J. Kelsey, S. Lucks, B. Schneier, M. Stay, D. Wagner, D. Whiting. Improved Cryptanalysis of Rijndael // *Fast Software Encryption*, 2000. – P. 213–230.
11. J. Toldinas, D. Rudzika, V. Štuikys, G. Ziberkas. Rootkit Detection Experiment within a Virtual Environment // *Electronics and Electrical Engineering*, 2009. – No. 8(104). – P. 63–68.
12. M. Andraşiu, Current approaches in modern cryptology, *Journal of Information Systems and Operations Management*, vol. 4, NO. 1, 2010.
13. R. Chandramouli. Battery power-aware encryption. // *ACM Transactions on Information and System Security (TISSEC)*, 2006. – vol. 9. – No. 2. – P. 162-180.
14. R. Damaševičius, V. E., Toldinas. Embedded program specialization for multiple criteria trade-offs // *Electronics and Electrical Engineering, Technologija*, 2008. – No. 8(88). – P. 9–14.
15. R. Toemeh, S. Arumugam. Breaking Transposition Cipher with Genetic Algorithm // *Electronics and Electrical Engineering. – Kaunas: Technologija*, 2007. – No. 7(79). – P 75–78.
16. S. Singhal, K. Narendra, L. Enhancement of Security through a Cryptographic Algorithm , *Pulkit International Journal of Computer and Electrical Engineering*, 2(5), No. 5, October, 2010, 1793-8163].
17. Website: <http://asciitablechart.com/index.html>, Extended ASCII character codes representation in Decimal, Octal, Hex and Binary.